STIC Search Report
EIC 2100

STIC Database Tracking Number: 97665

TO: Anh Ly
Location: 4A30
Art Unit : 2172
Wednesday, July 02, 2003

Case Serial Number: 09/479432

From: David Holloway
Location: EIC 2100
PK2-4B30
Phone: 308-7794

david.holloway@uspto.gov

Search Notes

Dear Examiner Ly,

Attached please find your search results for above-referenced case.
Please contact me if you have any questions or would like a re-focused search.

David

STIC EIC 2100 Search Request Form

(154) 97665

Today's Date: 06/30/03

What date would you like to use to limit the search?

Priority Date: 01/07/2000 Other:

Name Anh Ly

AU 2172 Examiner # 77831

Room # 4A30 Phone 306-4527

Serial # 09/479,432

Format for Search Results (Circle One):

(PAPER) DISK EMAIL

Where have you searched so far?

USP DWPI EPO JPO ACM IBM TDB

IEEE INSPEC SPI Other _____

Is this a "Fast & Focused" Search Request? (Circle One) YES (NO)

A "Fast & Focused" Search is completed in 2-3 hours (maximum). The search must be on a very specific topic and meet certain criteria. The criteria are posted in EIC2100 and on the EIC2100 NPL Web Page at http://ptoweb/patents/stic/stic-tc2100.htm.

What is the topic, novelty, motivation, utility, or other specific details defining the desired focus of this search? Please include the concepts, synonyms, keywords, acronyms, definitions, strategies, and anything else that helps to describe the topic. Please attach a copy of the abstract, background, brief summary, pertinent claims and any citations of relevant art you have found.

Metadata, Data warehouse
abstraction metadata
database metadata
transformation metadata
mapping metadata
translation (code, library, module)
~~wrapper~~ wrapper
mediator generator for classes
Data foundry metadata
UML = Unified Modeling language (p. 12 in spec)

STIC Searcher David Holloway Phone 308-7794

Date picked up 7-1-03 Date Completed 7-2-03

sira Search and Information Resources Administration

WEB RESULTS by Google (Showing Results 1 - 10 of 27)

1. untitled
... we discuss the design of a **mediator** specification **generator** (MSG) which attempts to derive **mediator** specifications from source **metadata** automatically ...
www.ipd.uka.de/~koenig/Publications/cia_ganzneu.ps - 0 B

2. untitled
... **Metadata** transformations that use higher order query language ... data items, a corresponde **generator**, and a ... and optimization in dis-tributed **mediator** systems. ...
research.microsoft.com/research/db/debull/99mar/ibm.ps - 0 B

3. untitled
... **Metadata** transformations that use higher order query ... the two schemas, a correspondence and a ... caching and optimization indistributed **mediator** systems ...
www.cs.toronto.edu/~miller/papers/HMN+99.ps - 0 B

4. Query Processing in Heterogeneous Systems
... **Generator**. **Wrapper**. ... some loose and partial schematic info may pay off tremendously. to "databasy" user/**mediator**/source interaction. ... 1. Schemas & **Metadata**. ...
db.cs.berkeley.edu/postmodern/papakonstantinou.ppt - 0 B

5. Information integration with attribution support for corporate ...
... around a central **mediator** whose integration ... a general-purpose **wrapper generator** tech developed ... Attribution information **Metadata** documenting information ...
context.mit.edu/~coin/publications/cikm99/cikm99.pdf - 0 B

6. Course information
... Buckets. URL pattern extractor and **generator**. COMP 631B. Web Information Retrieval. ... Us **Mediator**. (Query/Search/. Retrieval/Result). ... . **Metadata**/. Ontology. ...
course.cs.ust.hk/comp631b/1999Fall/Slides/05WebIR.ppt - 0 B

7. untitled
... **Metadata** Catalog. ... Query Interface Manager User Query Profile IDL **Generator**. ... services promoting source-independent query processing at the **mediator** level and ...
www.cse.ogi.edu/DISC/DIOM/online_pub/papers/97dcs-final.ps - 0 B

8. untitled
... Moreover, an automatic **generator** of Squirrel integrators has been ... is an implementation o **mediator** in the ... includes also information from a **metadata** store. ...

ftp.dis.uniroma1.it/pub/rosati/wp1-integration.ps - 0 B

9. untitled
... the inference agent and database manager provide the **mediator**. ... by the already mentione HTML **generator** of On2broker. ... and does not use any **metadata** approach ...
www.ubka.uni-karlsruhe.de/vvv/1998/wiwi/22/22.text - 61 KB

10. LESSONS LEARNED FROM APPLYING AI TO THE WEB 1. Introduction
... Fixpoint Procedure Lloyd-Topor **Transformation** Frame Logic to ... annotating web sources w processable **metadata**. ... a program (called **wrapper**) that extracts ...
math1.uibk.ac.at/users/c70385/ftp/paper/cij.pdf - 0 B

« Previous | Next »

Information integration with attribution support for corporate profiles

Thomas Lee
MIT Sloan School of Management
tlee@mit.edu

Melanie Chams
PwC Global Technology Centre
melanie.chams@us.pwcglobal.com

Robert Nado
PwC Global Technology Centre
bob.nado@us.pwcglobal.com

Stuart Madnick
MIT Sloan School of Management
smadnick@mit.edu

Michael Siegel
MIT Sloan School of Management
msiegel@mit.edu

ABSTRACT

The proliferation of electronically available data within large organizations as well as publicly available data (e.g. over the World Wide Web) poses challenges for users who wish to efficiently interact with and integrate multiple heterogeneous sources. This paper presents CI^3, a corporate information integrator, which applies XML as a tool to facilitate data mediation and integration amongst heterogeneous sources in the context of financial analysts creating corporate profiles. Sources include Lotus Notes, relational databases, and the World Wide Web. CI^3 applies a unified XML data model to automate integration. By preserving metadata about the source of each datum in the integrated result set, CI^3 supports source attribution. Users may trace the attribution metadata from the result back to the underlying sources and leverage their expertise in interpreting the data and, if necessary, use their judgment in assessing the authenticity and veracity of results. We present a functional overview of CI^3, its system architecture including the XML data model, and the integration procedures. We conclude by reflecting on lessons learned.

Keywords

Attribution, metadata, XML, data mediation, data integration

1. INTRODUCTION

As the amount of electronically available data and content continues to grow in scope and depth, the tremendous promise that the medium offers for improving productivity and facilitating the free flow of ideas is threatened by the weight of its own complexity. Information is distributed across machines around the world and stored in heterogeneous formats ranging from flat files to next generation, object-relational data warehouses. Moreover, across this expanse, the only consistent theme is the inconsistency of its semantics. More often than not, even within a single source, terms are defined and observed haphazardly. For example, what is a company name? What is the difference between "ATT," "AT&T", and "American Telephone and Telegraph?"

Mediation technologies integrate disparate information sources, hiding distribution and reconciling heterogeneity. The Context Interchange (COIN) Project at MIT is developing a model [8], a prototype [3], and tools [2] for the semantic integration of disparate (distributed and heterogeneous) information sources ranging from on-line databases to semi-structured Web services. From the perspective of a given data source, end-user, or intermediate application, context knowledge constitutes a declarative specification for how data is interpreted. By representing and reasoning about contexts, COIN's automated resolution of semantic conflicts enables transparent access to heterogeneous information sources [9].

In addition to the Context Interchange research project, several other projects [1, 7, 15] are developing mediation architectures. More recent strategies attempt to leverage the emergence of XML as a common, underlying data model to facilitate the integration of data from heterogeneous sources [10, 12].

The PwC-MIT Corporate Intelligent Information Integrator (CI^3) is an automated tool to facilitate disparate (heterogeneous and distributed) data integration. The original prototype was designed to gather corporate information to aide analysts in their daily activities at PricewaterhouseCoopers (PwC). CI^3 was developed to test a number of issues: accessing semi-structured data from the Web, query languages for the Web, heterogeneous data integration mixing relational data sources with semi-structured, Web-accessible content, and a means to test some of the capabilities and limitations of applying XML as a tool to support and facilitate data integration amongst heterogeneous sources. The approach is unique first in its emphasis on capturing metadata during query execution to document the

specific sources which contribute to each integrated result. Second, the prototype attempts to leverage the emerging XML standard while recognizing that the great preponderance of currently accessible resources will not provide native XML support.

In this paper, we describe the development of CI^3. We begin with a description of the application domain. The paper next details the functional description and the system architecture. We conclude by reflecting on lessons learned from CI^3 development and speculating on future work.

2. APPLICATION DOMAIN

On any given day, analysts at PricewaterhouseCoopers (PwC) scan newspapers, the wire services, stock ticker's, and other industry performance benchmarks in the course of providing services to existing customers as well as to identify potential clients. For a given company, a large amount of data is gathered from internal and external sources about the targeted company and its industry. Today, analysts must manually navigate the sea of available data: identify likely sources, translate data objectives into the corresponding source query language, and interpret and integrate results. Not only are some portions of these tasks simple and easily automated, but their repetitive nature also makes them prone to a degree of human error.

Based upon a stock ticker symbol or a company name, CI^3 draws upon a number of proprietary and public domain electronic resources to assemble company-specific business intelligence profiles. Profile data range from:

- News and general information
- Company directory information for contact information and background context
- Company officers and directors
- Company and industry specific performance indicators for absolute and comparative analysis
- Historical data to chart performance over time (both company and industry)
- Identify specific competitors for head-to-head comparisons
- Identify previous PwC engagements / expertise with the company in question
- Identify previous PwC engagements / expertise with industry / competitors

A number of similar sounding tools already exist. Free services such as CNNfn™ or Yahoo Financial™ provide news and statistical integration and aggregation. Fee-based services such as Hoover's™ add analyst reports and longitudinal data. The ability to customize profiles enables users to create standard forms for retrieving integrated company and industry-specific data. However, transparent access tends to blur the notion of distinct sources both in queries and results. The CI^3 *support tree* re-establishes the association between an integrated result set and its corresponding sources. Situations where users or applications might like to attribute the sources from which a particular datum is drawn include: evaluating data quality, measuring data timeliness, resolving conflicts, or seeking additional data [4, 11]. Note also that, unlike CNNfn, Yahoo Financial, and Hoover, CI^3 allows PwC, or any user of CI^3, to merge its own proprietary data with publicly available data.

Additionally, the task facing the analyst is to not only gather these different pieces of information, but also to draw, often complex, relationships between them. These connections draw upon a depth of expertise that is not available in marketed decision support systems. Although the current version does not fully implement such a capability, the output format provides a structure that could easily be interfaced to complex decision support systems.

3. FUNCTIONAL DESCRIPTION

As illustrated in Figure 1, CI^3 may be conceptualized in four layers: the data sources, the data access infrastructure, the CI^3 integration layer, and the users. CI^3 provides access to three different types of sources. The first type includes data on prior relationships between PwC and the company in question, expertise within PwC on the relevant industry, and any company-specific directory information or financial figures gleaned from prior PwC engagements. A database of longitudinal data on financial performance as derived from SEC filings constitutes a second source type. Type three primarily includes near real-time content: recent company and industry-specific news stories as well as up-to-the-minute figures on financial performance. Directory information and officers are also accessible from this source type.

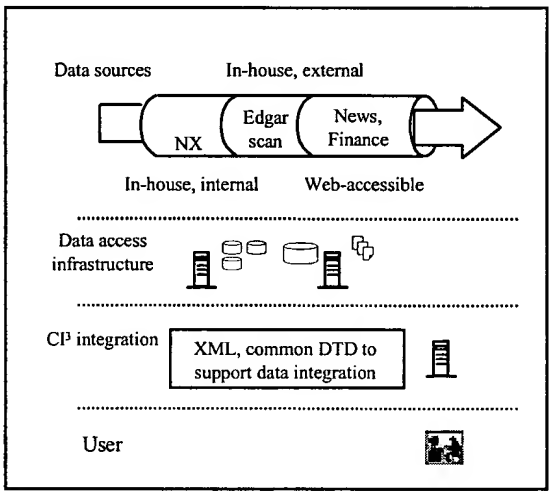

Figure 1. Functional description of CI^3

Although further described in the discussion on system architecture, the data source types loosely correspond to those sources that are internal to PwC, those that are maintained by PwC but exported to the world, and non-PwC content that is publicly accessible via the Internet. To reconcile this heterogeneity, CI^3 interposes an integrative layer between users and the source types.

These source types are all integrated in a single layer that accepts queries in the form of a company name or ticker symbol and returns an intelligence profile on the specified company. A single ticker-symbol or company name is transformed into respective sub-queries for the different sources and source types. Because the current use model is for a pre-formatted business intelligence sheet, all of the sub-queries are pre-specified. Future work, discussed below, aims to provide users with greater facility for manipulating sub-queries.

The integration layer is more than just a data integration step, however. It also serves the purpose of presenting the user with a single interface and access point to the wealth of information available on the Web and within PwC

4. SYSTEM ARCHITECTURE

The CI[3] system architecture is built around a central mediator whose integration strategy revolves around the use of the Extensible Markup Language (XML). Therefore, CI[3] serves not only as a tool to assist the PwC user but also as a testbed for exploring the use of XML as a data interchange standard. As illustrated in Figure 2, the system architecture employs three different access modalities to accommodate the three different types of sources described in the functional architecture. These different source types are unified through a single XML Document Type Definition DTD. After describing the three different access strategies, this section details the CI[3]'s underlying, XML-encoded data model and concludes by summarizing the integration procedure.

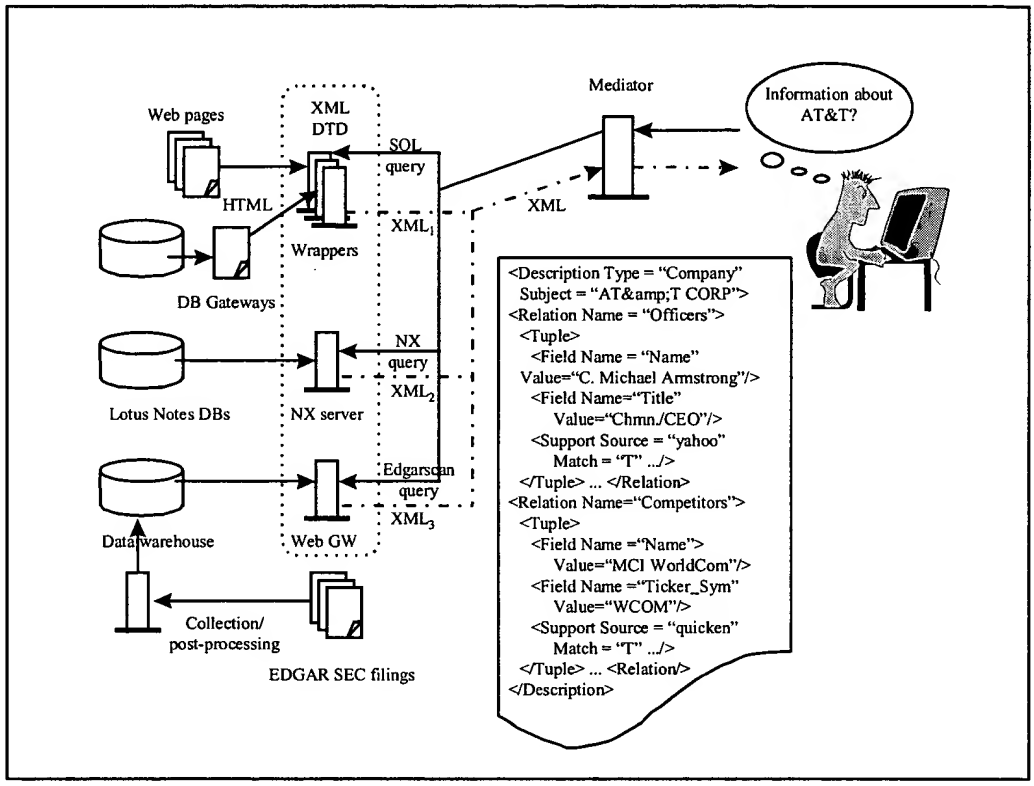

Figure 2. CI[3] system architecture

4.1 Source Types

The first source type consists of internal PwC Notes databases that provide general company information as well as information about prior PwC relationships with client companies. Notes databases contain documents that are formatted into a collection of fields that can contain discrete values as well as bodies of text, i.e. Notes databases are a kind of semi-structured information source. A proprietary search engine, called NX (Notes Explorer), has been developed at the PwC Global Technology Centre for retrieving information from a large collection of internal Notes databases [13]. NX supports a variety of search types, including full-text search as well as searches restricted to the contents of fields that have been meta-tagged as containing references to entities of specified types, such as people, companies, and skills. CI[3] makes use of a "company profile" search type that extracts structured, relational information about a specified company from the collection of underlying Notes databases. A single company profile query returns a variety of different types of information about a company.

The second source type is EdgarScan, a publicly accessible PwC system including a database of corporate facts and longitudinal data [5]. EdgarScan, reads and analyzes SEC EDGAR submissions, which are financial documents ("Semi-Structured" ASCII text files) filed by all public U.S. corporations with the U.S. Securities and Exchange Commission (SEC). EdgarScan pulls filings from SEC servers and parses them automatically to extract key financial tables and normalize those financials into a common format that is comparable across companies. A database is populated with the extracted financial data. EdgarScan exports a set of prespecified queries that are parameterized by company name or ticker symbol. Prespecified queries include retrieving an entire filing or specific, longitudinal financial performance metrics. Access to the database is provided via a number of avenues including a direct, JDBC interface and a CGI-based Web interface.

The third source type covers publicly accessible sites for which no non-Web mediated interface is exported. Instead, this category of queries, used by CI^3, relies upon a *wrapper* to access data from Web sites using a general-purpose *wrapper generator* technology developed at MIT [2]. A specification language utilizes patterns to describe the regularity of semi-structured sources, like Web pages, for extracting data. The language also defines a strategy for combining the data into a relational export schema. Given an SQL query, the wrapper considers the patterns for extracting the corresponding values, and creates and executes a query execution plan. The query execution plan describes the various steps for accessing the documents, collecting data from the documents into individual tables, and combining the intermediary results into the query answer. Data that is provided via Web-wrapping includes competitors, officers, and directory information.

4.2 Data model

The model is represented in XML as a tree rooted in a top-level Description element that corresponds to a single company. There were two principle reasons for selecting XML as the representation scheme. First, XML was developed as a standard for data interchange and industry trends suggest that XML may emerge as the representation of choice. Second, XML was developed around the stated objective of separating data presentation details from representation details. Although the focus of this paper is the underlying querying and integration, the use of XML to separate the interface presentation from the data representation has been key in having a flexible user interface (UI) and experimenting with alternative user interfaces.

Relational	CI^3 XML
Relation	Element <Relation>
Tuple (Row)	Element <Tuple>
Attribute (Column)	Element <Field>
Name of relation	Attribute "Name"
Name of attribute	Attribute "Name"
Value of attribute	Attribute "Value"

```
<Relation Name= "Competitors">
  <Tuple>
    <Field  Name ="Name"
         Value ="MCI WorldCom"> </Field>
    <Field  Name ="Ticker_Symbol"
         Value ="WCOM"> </Field>88
    <Support
    URL= "http://www.quicken.com/investments/
         comparison/ ?symbol=T "
        Date = "Mon Apr  5 11:20:14 US/Pacific 1999"
        Match = "T"
        Source = "quicken"
        Synopsis = ""
        Score = "1.000">
      </Support>
    </Tuple>
```

Table 1. Tuple for a competitor to AT&T (ticker = T)

The challenge in developing a common, underlying data model stemmed from differences in models among the three different source types which were reflected by the query languages and access modes. Given the differences, our intuition was to conceptually represent the model in a normal form where logically independent information is broken into separate CI^3 relations, thereby limiting redundancy and simplifying data integration. For example, a company's address was separated from the company's name and ticker symbol so that every time a company is listed (e.g. as a competitor to some other company), the address is not repeated.

Leveraging XML, CI^3 uses a tree-structured data model. However, flexibility in the XML specification left open the question of the distinction between XML elements, XML attributes, and the contents of XML elements. Because the intention was to represent a normal form of the relational model, the decision was made to represent the structural components of the relational model as elements. Relational attribute-value pairs are explicitly represented as XML attributes of a Field element (See the example in Table 1).

4.3 Attribution information

Metadata documenting information about specific sources, called a *Support*, is maintained at the tuple level (see Table 1). The support itself is a two-level OR-AND tree whose root is the Tuple. In the simplest case, the support for any result tuple instance consists of a single Support element which specifies the source attribution of that result. There are two cases where a Tuple element might contain multiple Support elements.

First, it is possible that two or more sources might independently, yet identically, populate a single tuple instance. In this eventuality, the support tree documents the alternative sources as siblings at the first level of the tree. Second, it is assumed that, by default, for any given tuple instance, all of the component fields are drawn from a single source. In some cases that assumption fails and more than one source contributes to the fields of a single tuple instance.

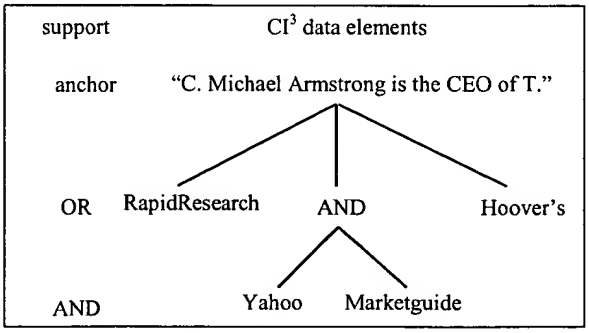

Figure 3. A support tree.

If more than one source is involved, contributing sources are represented as siblings at the second-level of the tree where their first-level parent is an AND node represented by the Combine element in CI^3. For Figure 3, one could independently query either RapidResearch or Hoover's to discover the CEO of AT&T Corp. Alternatively, the combination of data available from Yahoo and Marketguide could also return the AT&T CEO.

4.4 Integration procedure

Ultimately, CI^3 integration is the process of constructing an intelligence profile by combining CI^3 tuples into their respective CI^3 relations and returning the set of CI^3 relations. As noted earlier, CI^3 simultaneously executes a number of sub-queries

against different source types. Because each subquery submitted by the current version of CI^3 corresponds to complete tuples of one or more CI^3 relations, query processing in the form of selections, projections, or joins are not required.

However, because multiple sub-queries could populate a single CI^3 relation, a union operation is required. Moreover, as described in the system architecture, some native sources neither return XML nor correspond to the CI^3-data model. Integration, in these cases, therefore involves two intermediate steps before the final set of CI^3 relations is returned as an intelligence profile. Step one is the translation from the source query output into the CI^3 data model and the CI^3, XML-based, data representation. Step two is the process of taking the union over CI^3 tuples of the same CI^3 relation.

The union operation is straightforward. When Tuples from different sources are combined within the same relation, the associated support trees are merged as in Figure 3. It is assumed that each initial query is submitted to one of the three source types and corresponds to one or more of the initial base relations. Every result tuple is therefore an instance of a base relation and the composition of a company profile is only additive (e.g. there are no projections, joins, or other operations that would eliminate attributes). The remainder of this section will focus on the different translations required for the data models and representations that underlie the different source types.

The NX search engine used by CI^3 to access internal PwC Notes databases had already been designed to produce XML-formatted search results in order to separate presentation issues from the search engine output. The XML output for company profile searches was modified to conform to the CI^3 schema. This could be done without interfering with existing clients of the search engine because NX accepts a "client" search parameter that can be used to conditionally generate XML output according to different schemas. A single company profile search returns a Description element containing several different CI^3 relations as child elements. Each Support element contained in a tuple uses a URL attribute to reference the particular Notes document (hosted on a Domino server) that is the source of the tuple.

The EdgarScan system used by CI^3 to access company financial information and perform financial benchmarking is accessible through a CGI interface. A simple wrapper was also developed to make some EdgarScan query results available in XML. In practice, however, given the large amount of current and historical data extracted from EdgarScan and the fact that, in the CI^3 prototype, EdgarScan was the only source for this current and historical data, direct access to the underlying PwC EdgarScan database is used.

The wrappers which are used to extract relations from the Web, discussed earlier as the third source type, may not export relational schemas that are in a normal form. Therefore, for integration purposes, CI^3 must not only reduce the wrapper output into the appropriate, XML-formatted normal form, but also map the relations to the appropriate CI^3 Relation and CI^3 Field elements. Because the wrapper accepts relational queries, the mapping from the wrapper-relation to the CI^3 relation and between attribute names are driven through the query as though it were a view definition. Because the CI^3's XML DTD is structural, a simple procedure was written to restructure the relational table into the corresponding, XML-tagged output. Relations correspond to Relation elements, tuples to Tuple elements, etc.

5. LESSONS LEARNED

Development of the CI^3 prototype revealed a number of issues related to integration in general as well as some specific topics related to the use of XML for data interchange. For issues related to integration in general, both practical and conceptual lessons emerged. First, the importance of restructuring as a query language feature was emphasized. Because integration will likely involve multiple sources that rely upon different data models and query languages, facilities for transformation support are significant. As demonstrated in the case of the relational Web wrappers, a single relational query can simultaneously map attribute names as well as restructure data by means of views.

Second, CI^3 introduces the Support element as a means for associating each tuple with metadata about its corresponding base sources (See Table 1). The appropriate metadata set will vary depending upon the purpose of the reference [14] and is a discussion for another paper. In the context of CI^3, however, every support includes the name of the source, an access identifier to the source (e.g. a URL in the case of a Web reference), the query, the query terms which matched in a particular result tuple, and the last-modified date. This metadata is particularly useful because transparent data integration removes traditional cues for assessing the authenticity, veracity, and timeliness of data. Attribution provides analysts and other users with the means to inspect the original sources from which data is drawn. Moreover, although the current prototype is limited to publicly accessible or internally produced data sources, subsequent work could access proprietary or fee-based services. Future work might explore how integration and attribution strategies interface with rights-management or micro-payment schemes.

Also related to the problem of transparency over disparate sources is the existence of overlapping sources, which pose at least two kinds of problems: duplicates and contradictions. In the case of duplicates, the CI^3 data model's Support element accommodates multiple, corroborating sources. Contradictions, by contrast, raise many more subtle issues. Do contradictions truly signal errors, or might they point to fine-grained, underlying distinctions? For example, that two sources report different values for a company's *current* stock-price may simply reflect a difference in the time of reporting. Differences in a 52-week high or low may reflect different accounting conventions: 52-weeks from the time of reporting, the most recently-closed fiscal year, the most recently concluded calendar year, etc.

Finally, examples of fine-grained distinctions point to the broader problem of semantic heterogeneity in data integration. Homonyms, as in the case of different conventions for calculating a 52-week high or low, are only one class of semantic heterogeneity. Synonyms are a second such class. The challenge is further complicated because some definitions may be related although not identical. Definitions that subsume one another or intersect raise separate problems. [8]

In addition to posing a general problem, semantic heterogeneities also demonstrate an important challenge to

XML-enabled data integration. As a scaleable model capable of representing a global schema, XML offers a common framework for managing distributed data. In the simplest case, sources that rely upon the same Document Type Definition (DTD) are easily combined. Furthermore, XML Namespaces support the integration of sources with intersecting or disjoint DTDs. However, it is important to note that XML focuses on the syntax for representing content and does not provide any special facility for managing semantic differences between sources. Although developments such as XML-Schema and RDF hold promise, there is currently nothing to prevent data providers from adopting the same DTD but having different interpretations of the same terms.

Moreover, though two sources may provide the same underlying information, their DTDs may be structured in very different ways, providing little help to the task of integration.

Structural model	Ontology model
<!-- DTD for CI^3-XML --> <!ELEMENT Description (Relation*)> <!ELEMENT Relation (Tuple*)> <!ELEMENT Tuple (Field+, (Support | Combine)+)> <!ELEMENT Field EMPTY> <!ELEMENT Combine (Support , Support+)> <!ELEMENT Support EMPTY> <!-- attributes --> <!ATTLIST Description Type CDATA #REQUIRED Subject CDATA #REQUIRED ID ID #IMPLIED> <!ATTLIST Relation Name CDATA #REQUIRED> <!ATTLIST Field Name CDATA #REQUIRED Value CDATA #IMPLIED Category CDATA #IMPLIED> <!ATTLIST Support URL CDATA #REQUIRED DateCDATA #IMPLIED Match CDATA #IMPLIED Source CDATA #REQUIRED Synopsis CDATA #IMPLIED Score CDATA #IMPLIED >	<!-- DTD for CI^3-XML financial domain --> <!ELEMENT Company (Directory_info, Company_name, Ticker, Competitor*)> <!ELEMENT Directory_info (Address, Phone) > <!ELEMENT Company_name> <!ELEMENT Ticker (Ticker_symbol, Exchange)> <!ELEMENT Competitor EMPTY> <!ELEMENT Address (Street_address, City, State, Zip)> <!ELEMENT Ticker_symbol> <!ELEMENT Exchange> <!ELEMENT Street_address> <!ELEMENT City> <!ELEMENT State> <!ELEMENT Zip> <!ELEMENT Phone> <!-- attributes --> <!ATTLIST Company ID ID #IMPLIED> <!ATTLIST Competitor ID ID #IMPLIED>

Table 2. Different DTDs for the same underlying information.

For example, as illustrated by CI^3, a structural model of relations, tuples, and supports does not necessarily provide information on integrating content that corresponds to a DTD structured as a domain model (see Table 2).

6. CONCLUSION

We have described a system to integrate data from heterogeneous sources by simultaneously submitting real-time queries to multiple sources and integrating the results utilizing a common, underlying XML framework. The system was prototyped in the context of gathering business intelligence at PwC.

The integration of content from three different categories of sources utilizing different data models and query languages is demonstrated. First, the NX search engine is used to dynamically generate partial company profiles derived from a variety of internal PwC Notes databases. These databases include company directories, engagement records, contact information, best practices, and vendor information. Second, PwC maintains a publicly accessible data warehouse of SEC financial filings. Using custom data-extraction tools, PwC processes all SEC on-line filings and makes them available for searching and querying via the Web. Finally, based upon MIT semi-structured query wrapping technologies, CI^3 has relational query access to any number of semi-structured Web information sources. This includes a number of sites supporting general financial information and industry-by-industry comparisons.

The ultimate goal behind CI^3 is to gather data and assist in data analysis and identification of candidates for PwC professional services. To that end, future work will pursue this goal in several directions. First, as noted earlier, data integration is challenged by the semantic heterogeneities that span the myriad sources. Drawing upon research within the Context Interchange framework at MIT, one research direction will be to support the identification and resolution of inconsistent semantics. A second research direction will address the related problem of redundancy across the rich set of available sources. Different strategies for addressing problems of data duplication and data conflict include the identification and selection of reliable data sources at query-execution time, and the introduction of quality-assessment weights to provide users with indications of data veracity and source attribution. Third, linking CI^3 to decision support systems will enable analysts to draw more complex connections and conclusions from the data. Not only will the system be able to retrieve financial indicators, but it will also be able to reason about those variables, highlighting candidates for professional services. Finally, CI^3 is ultimately a system for human analysts. The current system translates a ticker-symbol or company name request into pre-specified queries against pre-selected sources. Future versions will support strategies for flexible query-answering over multiple sources.

7. REFERENCES

[1] Arens, Y. and C. Knobloch, "Planning and reformulating queries for semantically modelled multidatabase," In *Proc. of the Intl. Conf. on Information and Knowledge Management*, 1992.

[2] Bonnet, P. and S. Bressan, "Extraction and integration of data from semi-structured documents into business applications," In *Proc. of the Intl. Conf. on Industrial Applications of Prolog*, 1997.

[3] Bressan, S., C. Goh, K. Fynn, et al., "The context interchange mediator prototype," In *Proc. of the ACM SIGMOD Intl. Conf. on the Management of Data*, (demo track) 1997.

[4] Cui, Y., J. Widom and J. L. Wiener. "Tracing the lineage of view data in a warehousing environment. Technical Report, Stanford University Database Group, November 1997. http://www-db.stanford.edu /pub/papers/lineage-full.ps

[5] Ferguson, D., "Parsing financial statements efficiently and accurately using C and Prolog," *Conference on Practical Applications of Prolog*, February 1997.

[6] Fernandez, M., D. Florescu, A. Levy, and D. Suciu, "A query language for a web-site management system," *SIGMOD Record*, 26(3), 1997.

[7] Garcia-Molina, H., "The TSIMMIS approach to mediation: data models and languages," In *Proc. of the Conf. on Next Generation Information Technologies and Systems*, 1995.

[8] Goh, C., S. Madnick, and M. Siegel, "Context interchange: overcoming the challenges of large-scale interoperable database systems in a dynamic environment," In *Proc of the Intl. Conf. on Information and Knowledge Management*, 1994.

[9] Goh, C., S. Bressan, S. Madnick, and M. Siegel, "Context interchange: new features and formalisms for the intelligent integration of information" In *ACM Transactions on Information Systems*, July 1999.

[10] Goldman, R., J. McHugh, and J. Widom, "From semistructured data to XML: migrating the Lore data model and query language," *Technical Report, Stanford University, Department of Computer Science*, March 1999.

[11] Lee, T. and S. Bressan, "Multimodal integration of disparate information sources with attribution," In *ER97 Workshop on Information Retrieval and Conceptual Modeling*, 1997.

[12] McHugh, J, S. Abiteboul, R. Goldman, D. Quass, and J. Widom, "Lore: a database management System for semistructured data," *SIGMOD Record*, 26(3):54-66, September 1997.

[13] Nado, R. and S. Huffman, "Extracting entity profiles from semistructured information spaces," *SIGMOD Record*, Vol. 26, No. 4, December 1997, pp. 32-38; Also appeared *in Proceedings of the Workshop on Management of Semistructured Data*, May 1997, pp. 49-53.

[14] Rosenthal, A. and E. Sciore, "Helping data suppliers and consumers negotiate the details: a database view approach," submitted to *Conference on Cooperative Information Systems*, 1999.

[15] Tomasic, A., L. Raschid, and P. Valduriez, "Scaling heterogeneous database and the design of Disco," In *Proc. of the 16th Intl. Conf. on Distributed Computing Systems*, 1996.